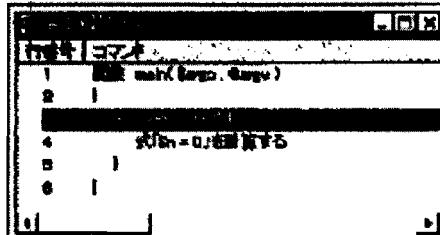


**DEVICE AND METHOD FOR EDITING AND PROVISION MEDIUM****Publication number:** JP2000222190**Publication date:** 2000-08-11**Inventor:** YANAGAWA FUMIHIKO; NISHIMURA TAKANORI;  
KUWAORI TAKAYUKI**Applicant:** SONY CORP**Classification:****- International:** G06F9/06; G06F3/00; G06F3/048; G06F9/06;  
G06F3/00; G06F3/048; (IPC1-7): G06F9/06; G06F3/00**- European:****Application number:** JP19990019597 19990128**Priority number(s):** JP19990019597 19990128**Report a data error here****Abstract of JP2000222190**

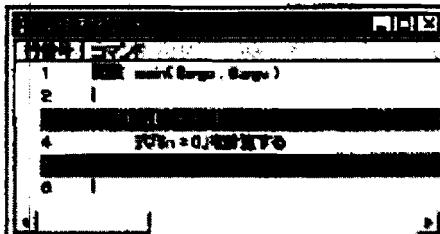
**PROBLEM TO BE SOLVED:** To maintain a correct grammar structure when a description having a specific grammar structure is edited as an object by retrieving a part paired with the part of the selected description to be edited and placing the part in a selected state when at least the part of the description to be edited is selected. **SOLUTION:** A 1st selection step for placing at least a part of the description to be edited in a selected state when the part is selected, a retrieval step for retrieving the part paired with the selected part of the description to be selected from the description to be edited, and a 2nd selection step for placing the retrieved part in a selected state are included. For example, when a 3rd line 'is YES' in a script (A) is selected is placed in the selected state, a 5th line ')where the closing parenthesis paired with the opening parenthesis is described is retrieved. Then the 5th line is also placed in a selected state (B).

(A)



```
1  sub main(Begin, End)
2  1
3  is YES
4  200n = 0.000000
5  )where the closing parenthesis paired with the opening parenthesis is described
6  1
```

(B)



```
1  sub main(Begin, End)
2  1
3  is YES
4  200n = 0.000000
5  )where the closing parenthesis paired with the opening parenthesis is described
6  1
```

Data supplied from the esp@cenet database - Worldwide

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開2000-222190

(P2000-222190A)

(43)公開日 平成12年8月11日 (2000.8.11)

(51)Int.Cl. <sup>7</sup>	識別記号	F I	テーマコード*(参考)
G 0 6 F 9/06	5 3 0	G 0 6 F 9/06	5 3 0 P 5 B 0 7 6
3/00	6 5 1	3/00	6 5 1 E 5 E 5 0 1

審査請求 未請求 請求項の数7 O.L (全20頁)

(21)出願番号 特願平11-19597

(22)出願日 平成11年1月28日 (1999.1.28)

特許法第64条第2項ただし書の規定により×印の部分及び図面第8図の一部は不掲載とした。

(71)出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72)発明者 柳川 史彦

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72)発明者 西村 孝則

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(74)代理人 100082131

弁理士 稲本 義雄

最終頁に続く

(54)【発明の名称】 編集装置および編集方法、並びに提供媒体

(57)【要約】

【課題】 スクリプトの編集を行う場合において、正しい文法構造を維持する。

【解決手段】 スクリプトにおける開き括弧「{」を含む行が選択されたときに、その行が選択状態とされ(図16(A))、その選択された行の開き括弧「{」と組になっている閉じ括弧「}」を含む行が検索される。そして、その検索された行も選択状態とされる(図16(B))。

行番号 マウス

1 関数 main(\$arg0, \$arg1)

2 [

3     if(\$arg0 == \$arg1)

4         式 \$a = \$b を計算する

5     ]

6 ]

(A)

行番号 マウス

1 関数 main(\$arg0, \$arg1)

2 [

3     if(\$arg0 == \$arg1)

4         式 \$a = \$b を計算する

5     ]

6 ]

(B)

**【特許請求の範囲】**

【請求項1】 所定の文法構造を有する記述を編集する編集装置であって、

前記編集対象の記述の少なくとも一部が選択されたときに、その一部分を選択状態とする第1の選択手段と、選択された前記編集対象の記述の一部と組になっている部分を、前記編集対象の記述から検索する検索手段と、前記編集対象の記述から検索された部分を選択状態とする第2の選択手段とを含むことを特徴とする編集装置。

【請求項2】 前記検索手段は、組になる括弧の一方を含む部分が選択されたときに、その括弧の他方を含む部分を検索することを特徴とする請求項1に記載の編集装置。

【請求項3】 前記検索手段は、選択された前記編集対象の記述の一部と組になっている部分を、行単位で検索することを特徴とする請求項1に記載の編集装置。

【請求項4】 前記選択状態とされた部分は、画面上において、他の部分と異なる表示状態とされることを特徴とする請求項1に記載の編集装置。

【請求項5】 前記編集対象の記述は、所定の処理手順を指示するスクリプトであることを特徴とする請求項1に記載の編集装置。

【請求項6】 所定の文法構造を有する記述を編集する編集方法であって、

前記編集対象の記述の少なくとも一部が選択されたときに、その一部分を選択状態とする第1の選択ステップと、

選択された前記編集対象の記述の一部と組になっている部分を、前記編集対象の記述から検索する検索ステップと、

前記編集対象の記述から検索された部分を選択状態とする第2の選択ステップとを含むことを特徴とする編集方法。

【請求項7】 所定の文法構造を有する記述を編集する処理を、コンピュータに行わせるためのコンピュータプログラムを提供する提供媒体であって、

前記編集対象の記述の少なくとも一部が選択されたときに、その一部分を選択状態とする第1の選択ステップと、

選択された前記編集対象の記述の一部と組になっている部分を、前記編集対象の記述から検索する検索ステップと、

前記編集対象の記述から検索された部分を選択状態とする第2の選択ステップとを含むコンピュータプログラムを提供することを特徴とする提供媒体。

**【発明の詳細な説明】****【0001】**

【発明の属する技術分野】 本発明は、編集装置および編集方法、並びに提供媒体に関し、特に、所定の文法構造を有する記述を対象とした編集を行う場合において、正

しい文法構造を維持することができるようにする編集装置および編集方法、並びに提供媒体に関する。

**【0002】**

【従来の技術】 近年、携帯型の電子機器が開発され、CPU (Central Processing Unit) を搭載した高度な電子機器が普及し始めている。この携帯型の電子機器である携帯型のパーソナルコンピュータは、本体とそのLCD (Liquid Crystal Display) 等の表示部を備えており、この表示部は、本体に対してヒンジを介して開閉できるようになっている。このような携帯型のパーソナルコンピュータは、ノート型のパーソナルコンピュータと称されている。

【0003】 パーソナルコンピュータ（以下、適宜、パソコンという）においては、例えば、Windows95やWindows98（いずれも商標）等に代表されるOS（Operating System）が起動された後、所望のプログラムが起動され、所望の処理を行うことができるようになる。例えば、電子メールの授受を行いたい場合、OSを起動し、さらに電子メールおよびダイアルアップ等のアプリケーションプログラムを起動することにより、自分宛のメールが届いていないかチェックすることができる状態となる。

【0004】 しかしながら、ユーザがパソコンにおいて電子メールを即座にみたいという希望がある。また、電子メールの機能に限らず、パソコンが備える機能を瞬時に利用したいという希望がある。このような要請に応えるために本出願人は、例えば、特願平9-318341や特願平10-184355において、所定のプログラムを1回の操作で起動させるための操作キー（後述する図3や図9に示すPPK9）を備えたパソコンを提案している。このようなパソコンによれば、操作キーを1度だけ操作すれば、予め設定されたプログラム、例えば、電子メールおよびダイアルアップ等のプログラムが自動的に起動し、自分宛のメールを取得することができる。

【0005】 ところで、パソコンに、ある一連の処理を行わせるためには、マウスやキーボード等を、その一連の処理に対応して操作する必要がある。即ち、例えば、上述のように、電子メールをチェックしたい場合には、ユーザは、パソコンのOSの起動（電源を投入、あるいはサスペンション状態を解除）、ダイアルアップのアプリケーションプログラムの起動、電子メールのアプリケーションプログラムの起動、メールサーバにアクセスするためのアカウントおよびパスワードの入力等を行い、これにより、パソコンに、電子メールを取得するための一連の処理を行わせなければならない。

【0006】 上述の操作キーに、電子メールを取得するための一連の処理手順（ユーザの操作手順と等価）を対応付けておけば、ユーザは、操作キーを操作するだけで、電子メールのチェックを行うことができるが、操作キーと、電子メールを取得するための一連の処理手順と

の対応付けを固定にしたのでは、パソコンに、電子メールの取得以外の処理を行わせたいユーザの要望に応えることができない。従って、操作キーと、ある一連の処理手順とは、可変に対応付けることができるようになるのが望ましい。例えば、本件出願人である株式会社ソニー製のパソコン×××××××××は、そのような可変の対応付けを行うことができるようになされている。

【0007】ところで、操作キーと、ある一連の処理手順との対応付けを行う場合においては、一連の処理手順を記述しておく必要がある。この一連の処理手順を記述するためのツールとしては、例えば、上述のパソコン××××にプレインストールされているエー・アイ・ソフト株式会社製のScriPlayerPRO32（商標）というアプリケーションプログラムがある。なお、ScriPlayerPRO32については、エー・アイ・ソフト株式会社のホームページ（<http://www.aisoft.co.jp/>）等に、その詳細が開示されている。

【0008】ユーザは、処理手順を記述する場合、例えば、上述のパソコン××××において、Windows95を起動させ、そのデスクトップにおける「スタート」ボタンをクリックして、メニューの中にある「プログラム」の中の「ScriPlayerPRO32 Lite 2」から「ScriPlayerPRO32」を選択する。これにより、ScriPlayerPRO32が起動される。

【0009】図1は、ScriPlayerPRO32の編集画面を示している。

【0010】ScriPlayerPRO32の編集画面は、ScriPlayerPRO32に対して、各種の指示を与えるためのツールバーや、一連の処理手順の記述（以下、適宜、スクリプトという）がされるスクリプト編集ウインドウ等から構成されている。

【0011】ScriPlayerPRO32では、編集画面におけるツールバーの「記録」ボタンがクリックされると、処理手順の記録が開始される。即ち、「記録」ボタンをクリックした後のユーザの操作は記録され、その操作に対応するスクリプトが、スクリプト編集ウインドウに書き込まれる。この記録は、ツールバーの「停止」ボタンがクリックされるまで続けられる。そして、「停止」ボタンがクリックされると、ユーザの操作の記録が停止され、「記録」ボタンがクリックされた直後から、「停止」ボタンがクリックされる直前までのユーザの操作により行われた処理に対応するスクリプトが、スクリプト編集ウインドウに記述される。

【0012】ここで、図2に、スクリプトが記述された状態のスクリプト編集ウインドウを示す。同図に示すように、スクリプトについては、ユーザが処理の内容を容易に認識することができるような記述が可能となっている。

【0013】スクリプトの記述後、ツールバーの実行ボタンをクリックすると、ScriPlayerPRO32では、そのス

クリプトが解釈、実行される。即ち、これにより、「記録」ボタンがクリックされた直後から、「停止」ボタンがクリックされる直前までのユーザの操作に対応した一連の処理が行われる。ここで、このように、スクリプトは解釈されて実行されるから、コンピュータプログラムの一種であると考えることができる。

【0014】また、スクリプトは、ファイルとして保存しておくことができ、そのようにファイルとして保存されたスクリプトと、上述の操作キーとを対応付けておくことにより、操作キーを操作するだけで、スクリプトとして記述された処理、即ち、ユーザが所望する処理を、パソコンに、即座に行わせることができる。

【0015】

【発明が解決しようとする課題】ところで、スクリプト編集ウインドウには、上述したように、ユーザが実際に行った操作に対応する処理手順を、スクリプトとして反映させることができる他、ユーザが、スクリプトを、直接入力することもできる。さらに、スクリプト編集ウインドウに記述されたスクリプトは、ユーザが編集することもできる。

【0016】しかしながら、スクリプトは、ScriPlayerPRO32が解釈して実行するものであるから、ScriPlayerPRO32が解釈するための一定の文法構造を有している。即ち、スクリプトの記述には、例えば、()や{}などの括弧が用いられるが、括弧については、開き括弧（「（」や「{」など）と、閉じ括弧（「）」や「}」など）とは、必ず組で用いられる。

【0017】従って、スクリプトの編集において、例えば、開き括弧を消去し、それと組になる閉じ括弧を消去しないと、スクリプトの編集結果は、文法的に誤った構造となり、ScriPlayerPRO32が解釈することができなくなる。

【0018】即ち、例えば、いま、

```
1: 関数f() {
2: もし○○ならば {
3:   ××する
4: }
5: }
```

といったスクリプトがあった場合に、開き括弧「（」がある第2行目だけ削除すると、

```
1: 関数f() {
2:   ××する
3: }
```

となる。この場合、削除した開き括弧「（」と組になっていた第3行目の閉じ括弧「）」が余り、文法的に誤ったスクリプトとなる。

【0019】上述のような簡単なスクリプトの場合はともかく、複雑なスクリプトを編集する場合には、このような事態が頻発することが予想される。

【0020】本発明は、このような状況に鑑みてなされたものであり、所定の文法構造を有する記述を対象とした編集を行う場合において、正しい文法構造を維持することができるようにするものである。

【0021】

【課題を解決するための手段】本発明の編集装置は、編集対象の記述の少なくとも一部が選択されたときに、その一部分を選択状態とする第1の選択手段と、選択された編集対象の記述の一部と組になっている部分を、編集対象の記述から検索する検索手段と、編集対象の記述から検索された部分を選択状態とする第2の選択手段とを含むことを特徴とする。

【0022】本発明の編集方法は、編集対象の記述の少なくとも一部が選択されたときに、その一部分を選択状態とする第1の選択ステップと、選択された編集対象の記述の一部と組になっている部分を、編集対象の記述から検索する検索ステップと、編集対象の記述から検索された部分を選択状態とする第2の選択ステップとを含むことを特徴とする。

【0023】本発明の提供媒体は、編集対象の記述の少なくとも一部が選択されたときに、その一部分を選択状態とする第1の選択ステップと、選択された編集対象の記述の一部と組になっている部分を、編集対象の記述から検索する検索ステップと、編集対象の記述から検索された部分を選択状態とする第2の選択ステップとを含むコンピュータプログラムを提供することを特徴とする。

【0024】本発明の編集装置および編集方法、並びに提供媒体においては、編集対象の記述の少なくとも一部が選択されたときに、その一部分が選択状態とされ、その選択された編集対象の記述の一部と組になっている部分が、編集対象の記述から検索される。そして、編集対象の記述から検索された部分も選択状態とされる。

【0025】

【発明の実施の形態】図3乃至図8は、本発明を適用した携帯型パソコンコンピュータ（パソコン）の構成例を表している。このパソコン1は、ミニノート型のパソコンと呼ばれるもので、本体2と、本体2に対して開閉自在とされている表示部3とから構成されている。図3は、表示部3を本体2に対して開いた状態を示す外観斜視図、図4は、図3の平面図、図5は、表示部3を本体2に対して閉塞した状態を示す左側側面図、図6は、表示部3を本体2に対して180度開いた状態を示す右側側面図、図7は、図5の正面図、図8は、図6の底面図である。

【0026】本体2には、各種の文字や記号などを入力するとき操作されるキーボード4、（マウス）カーソルを移動させるときなどに操作されるトラックポイント（商標）5が、その上面に設けられている。また、本体2の上面には、音を出力するスピーカ8と、表示部3に設けられているCCDビデオカメラ23で撮像するとき操

作されるシャッタボタン10がさらに設けられている。

【0027】表示部3の上端部には、ツメ13が設けられており、図5に示すように、表示部3を本体2に対して閉塞した状態において、ツメ13に対向する位置における本体2には、ツメ13が嵌合する孔部6が設けられている。本体2の前面には、スライドレバー7が前面に平行に移動可能に設けられており、スライドレバー7は孔部6に嵌合したツメ13と係合してロックし、またロック解除することができるようになっている。ロックを解除することにより、表示部3を本体2に対して回動することができる。ツメ13の隣りには、マイクロホン24が取り付けられている。このマイクロホン24は、図8にも示すように、背面からの音も収音できるようになされている。

【0028】本体2の正面にはまた、プログラマブルパワーキー（PPK）9が設けられている。本体2の右側面には、図6に示すように、排気孔11が設けられており、本体2の前面下部には、図7に示すように、吸気孔14が設けられている。さらに、排気孔11の右側には、PCMCIA（Personal Computer Memory Card International Association）カード（PCカード）を挿入するためのスロット12が設けられている。

【0029】表示部3の正面には、画像を表示するLCD（Liquid Crystal Display）21が設けられており、その上端部には、撮像部22が、表示部3に対して回動自在に設けられている。すなわち、この撮像部22は、LCD21と同一の方向と、その逆の方向（背面の方向）との間の180度の範囲の任意の位置に回動することができるようになされている。撮像部22には、CCD（Charge Coupled Device）ビデオカメラ23が取り付けられている。

【0030】表示部3の下側の本体側には、電源ランプPL、電池ランプBL、メッセージランプML、その他のLEDよりなるランプが設けられている。なお、図5に示す符号40は、本体2の左側面に設けられた電源スイッチであり、図7に示す符号25は、CCDビデオカメラ23のフォーカスを調整する調整リングである。さらに、図8に示す符号26は、本体2内に増設メモリを取り付けるための開口部を被覆する蓋であり、符号41は、蓋26のロックツメを外すためのピンを挿入する小孔である。

【0031】次に、図9は、パソコン1の内部の構成を表している。内部バス51には、図9に示すように、CPU（Central Processing Unit）52、必要に応じて挿入されるPCカード53、RAM（Random Access Memory）54、およびグラフィックチップ81が接続されている。この内部バス51は、外部バス55に接続されており、外部バス55には、ハードディスクドライブ（HDD）56、I/O（入出力）コントローラ57、キーボードコントローラ58、トラックポイントコントローラ59、サウンドチップ60、LCDコントローラ83、モ

デム（MODEM）50などが接続されている。

【0032】CPU52は、各機能を統括するコントローラであり、RAM54等にロードされた各種のプログラムを実行することにより、各種の処理を行う。PCカード53は、オプションの機能を付加するとき適宜装着される。

【0033】RAM54の中には、パソコン1の電源が投入（あるいは、サスPEND状態が解除）されると、例えば、Windows95やWindows98などのOS（基本プログラム）が、HDD56から転送され、CPU52により実行される。これにより、OSが起動される。さらに、OSの起動が完了した後、必要に応じて、所定のアプリケーションプログラムが、HDD56からRAM54に転送され、CPU52により実行される。

【0034】外部バス55側のHDD56には、OSの他、各種のアプリケーションプログラムが記憶されており、必要に応じて、RAM54に転送され、上述したように、CPU52により実行される。ここで、本実施の形態においては、HDD56には、実行エンジンプログラムや、スマートスクリプト（SmartScript（商標））エディタプログラム、その他のアプリケーションプログラムが記憶されている。

【0035】なお、実行エンジンプログラムは、スクリプトを解釈、実行するインタプリタ等を含む実行エンジンとして機能するようになされている。また、スマートスクリプトエディタプログラムは、スクリプトを編集等するスマートスクリプトエディタとして機能するようになされている。実行エンジンおよびスマートスクリプトエディタの詳細については、後述する。

【0036】I/Oコントローラ57は、マイクロコントローラ61を有し、このマイクロコントローラ61には、I/Oインターフェース62が設けられている。このマイクロコントローラ61は、I/Oインターフェース62、CPU63、RAM64、ROM69が相互に接続されて構成されている。このRAM64は、キー入力ステイタスレジスタ65、LED（発光ダイオード）制御レジスタ66、設定時刻レジスタ67、レジスタ68を有している。設定時刻レジスタ67は、ユーザが予め設定した時刻になると起動シーケンス制御部76の動作を開始させる際に利用される。レジスタ68は、予め設定された操作キーの組み合わせと、起動すべきアプリケーションプログラム、またはスクリプトの対応を記憶するもので、その記憶された操作キーの組み合わせがユーザにより入力されると、その記憶されたアプリケーションプログラムまたはスクリプト（スクリプトを実行する実行エンジン）が起動されることになる。

【0037】キー入力ステイタスレジスタ65には、ワンタッチ操作用のプログラムブルパワーキー（PPK）9が押されると、操作キーフラグが格納されるようになっている。LED制御レジスタ66は、レジスタ68に記憶

されたアプリケーションプログラムの立ち上げ状態を表示するメッセージランプMLの点灯を制御するものである。設定時刻レジスタ67は、所定の時刻を任意に設定することができるものである。

【0038】なお、このマイクロコントローラ61には、バックアップ用のバッテリ74が接続されており、各レジスタ65、66、67の値は、本体2の電源がオフとされている状態においても保持されるようになっている。

【0039】マイクロコントローラ61内のROM69の中には、ウェイクアッププログラム、キー入力監視プログラム、LED制御プログラムが予め格納されている。このROM69は、フラッシュメモリである、例えばEEPROM（electrically erasable andprogrammable read only memory）で構成されている。さらに、マイクロコントローラ61には、常時現在時刻を計時するRTC（Real-Time Clock）75Aが接続されている。

【0040】ここで、ROM69の中のウェイクアッププログラムは、RTC75Aから供給される現在時刻データに基づいて、設定時刻レジスタ67に予め設定された時刻になったかどうかをチェックして、設定された時刻になると、所定の処理（またはプログラム）などの起動をするプログラムである。キー入力監視プログラムは、PPK9が利用者により押されたかどうかを常時監視するプログラムである。LED制御プログラムは、メッセージランプMLの点灯を制御するプログラムである。

【0041】ROM69には、さらにBIOS（Basic Input/Output System）（基本入出力システム）のプログラムが書き込まれている。このBIOSによって、OSやアプリケーションソフトウェアと周辺機器（ディスプレイ、キーボード、ハードディスクドライブなど）の間でのデータの受け渡し（入出力）が制御される。

【0042】外部バス55に接続されているキーボードコントローラ58は、キーボード4からの入力をコントロールする。トラックポイントコントローラ59は、トラックポイント5の入力を制御する。

【0043】サウンドチップ60は、マイクロホン24からの入を取り込み、あるいは内蔵スピーカ8に対して音声信号を供給する。

【0044】モデム50は、公衆電話回線90、インターネットサービスプロバイダ91を介して、インターネットなどの通信ネットワーク92、メールサーバ93、WWW（World Wide Web）サーバ94などと通信を行うための通信制御を行う。

【0045】内部バス51に接続されているグラフィックチップ81には、CCDビデオカメラ23で取り込んだ画像データが、処理部82で処理された後、入力されるようになされている。グラフィックチップ81は、処理部82を介してCCDビデオカメラ23より入力されたビデオデータ等を、内蔵するVRAM81Aに記憶し、適宜、

これを読み出して、LCDコントローラ83に出力する。LCDコントローラ83は、グラフィックチップ81より供給された画像データをLCD21に出力し、表示させる。バックライト84は、LCD21を後方から照明するようになされている。

【0046】電源スイッチ40は、電源をオンまたはオフするとき操作される。半押しスイッチ85は、シャッターボタン10が半押し状態にされたときオンされ、全押しスイッチ86は、シャッターボタン10が全押し状態にされたときオンされる。反転スイッチ87は、撮像部22が180度回転されたとき（CCDビデオカメラ23がLCD21の反対側を撮像する方向に回転されたとき）、オンされるようになされている。

【0047】以上のように構成されるパソコン1では、HDD56に記憶されたスマートスクリプトエディタプログラムや、実行エンジンプログラムが実行されることによって、上述のScriPlayerPRO32と同様に、スクリプトを記述し、さらに実行することができるようになされており、これにより、ある一連の処理を自動的に行うことができるようになされている。

【0048】即ち、ユーザは、スクリプトを記述する場合、パソコン1において、OSであるWindows95等を起動させ、そのデスクトップにおける「スタート」ボタンをクリックして、メニューの中にある「プログラム」の中のスマートスクリプトエディタを選択する。この場合、HDD56に記憶されたスマートスクリプトエディタプログラムが、RAM54にロードされて実行され、これにより、スマートスクリプトエディタが起動される。

【0049】スマートスクリプトエディタが起動されると、例えば、図10に示すような、スマートスクリプトエディタのメインウインドウ131が表示される。

【0050】スマートスクリプトエディタのメインウインドウ131においては、ScriPlayerPRO32と同様に、ユーザの操作に対応するスクリプトが記述されるようになされており、また、スクリプトは、ファイルとして、HDD56に保存しておくことができるようになされている。

【0051】なお、スマートスクリプトエディタでは、ユーザが、スクリプトを直接入力し、また編集等することができるようにもなされており、その詳細については後述する。

【0052】ファイルとされたスクリプトを実行する場合においては、デスクトップにおける「スタート」ボタンをクリックして、メニューの中にある「プログラム」の中のスマートスクリプトインタプリタを選択する。この場合、HDD56に記憶された実行エンジンプログラムが、RAM54にロードされて実行され、これにより、実行エンジンが起動される。

【0053】実行エンジンが起動されると、デスクトップには、例えば、図11に示すように、解釈して実行す

べきスクリプトのファイルを選択するための「ファイルを開く」ウインドウ101が表示される。そして、ユーザが、スクリプトのファイルを選択して、「開く」ボタン102をクリックすると、実行エンジンによって、選択されたスクリプトが解釈、実行される。即ち、これにより、選択されたスクリプトに対応する一連の処理が行われる。

【0054】ここで、図12は、図11においてデスクトップの下部に表示されているWindows95のタスクバー103を示している。

【0055】図12のタスクバー103において、その左端には、「スタート」ボタン111が配置されている。「スタート」ボタン111は、各種のアプリケーションを起動するときや、Windows95を終了するときなどに操作される。「スタート」ボタンの右隣には、いわゆるクイック起動するアプリケーションに対応するアイコン112が配置されており、アイコン112がクリックされると、そのクリックされたものに対応するアプリケーションが起動される。さらに、その右隣には、タスクボタン113が配置される。タスクボタン113は、現在開いているウインドウに対応する数だけ表示される。従って、アプリケーションを起動してウインドウを開けば、そのアプリケーションに対応するタスクボタン113が表示され、アプリケーションを終了し、ウインドウを閉じれば、そのアプリケーションに対応するタスクボタン113は消去される。なお、ウインドウを最小化すると、ウインドウは、デスクトップから消去され、タスクボタン113として残り、この場合、タスクボタン113をクリックすると、再び、ウインドウが、デスクトップに表示される。

【0056】タスクバー103の右端には、タスクトレイ114が配置されている。タスクトレイ114は、前述したように、起動しているアプリケーションに対応するアイコンを表示する領域であり、特に、常に起動しておくようなアプリケーションは、タスクトレイ114にアイコンとして表示される。なお、図12において、タスクトレイ114の最も右にあるアイコン114Aが、実行エンジンに対応するもので、このアイコン114Aは、実行エンジンの起動後、それが終了されるまで、タスクトレイ114に表示された状態となる。

【0057】次に、図13のフローチャートを参照して、実行エンジンに対応するアイコンを表示するアイコン表示処理について説明する。なお、このアイコン表示処理は、実行エンジンによって行われる。

【0058】実行エンジンが起動されると、表示処理では、まず最初に、ステップS1において、インタプリタが起動しているかどうかが判定される。即ち、実行エンジンは、スクリプトを解釈して実行するインタプリタ（スクリプトを解釈する解釈部と、その解釈結果にしたがって、スクリプトを実行する実行部とからなるインタ

プリタとして機能するプログラム)を有しており、そのインタプリタが起動しているかどうかが判定される。ステップS1において、インタプリタが起動していると判定された場合、ステップS2に進み、タイマが起動される。

【0059】即ち、実行エンジンは、時間を計時するタイマ(として機能するプログラム)を有しており、ステップS2では、そのタイマが起動される。そして、タイマは、所定の時間を計時すると、その計時を終了し、インタプリタの実行状態を検出する状態検出部(として機能するプログラム)を呼び出す。即ち、実行エンジンは、インタプリタの実行状態を検出する状態検出部を有しており、この状態検出部では、ステップS3において、インタプリタが、例えば、動作状態、エラー状態、停止状態のいずれの状態にあるのかが検出される。さらに、ステップS3では、その検出結果が、アイコン選択部(として機能するプログラム)に渡される。

【0060】即ち、実行エンジンは、複数のアイコンの中から1つを選択するアイコン選択部を有しており、このアイコン選択部では、ステップS3において、状態検出部からのインタプリタの実行状態に対応したアイコンが選択される。

【0061】具体的には、アイコン選択部は、動作状態用のアイコン、エラー状態用のアイコン、停止状態用のアイコンを有しており、状態検出部からのインタプリタの実行状態に対応したアイコンを選択する。

【0062】そして、ステップS4に進み、アイコン選択部で選択されたアイコンが、タスクトレイ114に表示される。即ち、実行エンジンは、タスクトレイ114にアイコンを表示させるための表示制御部(として機能するプログラム)を有しており、表示制御部は、ステップS4において、アイコン選択部が選択したアイコンを、タスクトレイ114に表示させる。

【0063】その後、ステップS1に戻り、ステップS1において、インタプリタが起動していないと判定されるまで、ステップS1乃至S4の処理を繰り返す。

【0064】一方、ステップS1において、インタプリタが起動していないと判定された場合、即ち、実行エンジンが終了された場合、表示処理を終了する。

【0065】ここで、アイコン選択部は、動作状態用のアイコンとして、所定のアニメーションを構成する複数のアイコンを有しており、状態検出部からのインタプリタの実行状態を受信するごとに、その実行状態が動作状態である限り、動作状態用の複数のアイコンを所定の順番で選択するようになされている。従って、インタプリタの実行状態が動作状態となっている間は、タイマによって所定の時間が計時されるごとに、動作状態用の複数のアイコンが所定の順番で選択され、その結果、タスクトレイ114においては、図14に示すように、インタプリタが動作状態であることを表すアイコン114Aの

アニメーションが表示される。なお、図14においては、ベルトコンベアによって、立方体の荷物が運ばれているアニメーションのアイコン114Aが表示されている。

【0066】また、アイコン選択部は、エラー動作状態用または停止状態用のアイコンとして、それぞれ異なるアニメーションを構成する複数のアイコンを有しており、インタプリタの実行状態がエラー状態である場合には、エラー状態用の複数のアイコンを所定の順番で選択し、インタプリタの実行状態が停止状態である場合には、停止状態用の複数のアイコンを所定の順番で選択するようになっている。従って、インタプリタの実行状態がエラー状態となっている間は、タイマによって所定の時間が計時されるごとに、エラー状態用の複数のアイコンが所定の順番で選択され、その結果、タスクトレイ114においては、インタプリタがエラー状態であることを表すアイコン114Aのアニメーションが表示される。また、インタプリタの実行状態が停止状態となっている間は、タイマによって所定の時間が計時されるごとに、停止状態用の複数のアイコンが所定の順番で選択され、その結果、タスクトレイ114においては、インタプリタが停止状態であることを表すアイコン114Aのアニメーションが表示される。

【0067】従って、インタプリタの実行状態に対応したアニメーションで、アイコン114Aが表示されるので、ユーザは、そのアイコンを一目見ただけで、インタプリタの実行状態、即ち、インタプリタが動作状態、エラー状態、停止状態のうちのいずれの状態にあるのかを、容易に認識することができる。

【0068】なお、アニメーションのアイコンは、インタプリタが動作状態にあるときのみ表示し、インタプリタがエラー状態や停止状態にあるときは、アニメーションを止めて、エラー状態や停止状態を表す特定の(静的な)アイコンを表示するようにしても良い。さらに、エラー状態や停止状態の場合は、動作状態の場合に表示されるアイコンのアニメーションを停止させるようにしても良い。

【0069】また、インタプリタの実行状態に対応したアニメーションでのアイコンの表示は、タスクトレイ114において行う他、例えば、デスクトップ上のウインドウにおいて行うようにすることも可能である。即ち、例えば、インタプリタが起動された場合には、例えば、図15に示すようなダイアログボックス121を、デスクトップに表示し、そのダイアログボックス121上のアイコン121Aを、上述したようなインタプリタの実行状態に対応したアニメーションで表示するようにすることが可能である。ここで、図15において、最小化ボタン122がクリックされると、ダイアログボックス121は最小化され、タスクトレイ114に、アイコン114Aとして格納される。また、クローズボタン123

またはキャンセルボタン124がクリックされると、実行エンジンは終了され、ダイアログボックス121はクローズされる。

【0070】さらに、本実施の形態においては、スクリプトを解釈して実行するインタプリタの実行状態に対応したアニメーションで、アイコンを表示させるようにしたが、その他のアプリケーションプログラム（例えば、ドライバや、所定のツール）の実行状態に対応したアニメーションで、アイコンを表示させることも可能である。

【0071】また、前述したように、スクリプトは、PPK9と対応付けておき、PPK9が操作されたときに、それに対応付けられているスクリプトを、実行エンジンに実行させることも可能である。この場合、ユーザが、PPK9を操作するだけで、パソコン1に所定の一連の処理を行わせることができる。なお、PPK9については、前述したように、本出願人が先に出願した、特願平9-318341や特願平10-184354に、その詳細が開示されているので、これらを参照されたい。

【0072】次に、スマートスクリプトエディタにおいては、上述したように、ユーザが、スクリプトを編集することができるようになされている。

【0073】スクリプトに限らず、各種の言語によるプログラムや、文章等の編集においては、一般に、コピーや、切り取り、削除等が多用されるが、これらの操作を行う場合には、そのコピーや、切り取り、削除を行う部分を選択し、選択状態とする必要がある。

【0074】スマートスクリプトエディタでは、スクリプトのある行が選択されたときに、その行を選択状態とされる他、その選択された行と組になっている行が、スクリプトから検索され、その検索された行も選択状態とされるようになされている。

【0075】即ち、例えば、いま、図16（A）に示すようなスクリプトにおいて、その第3行目「もしYESならば{」が選択状態とされた場合には、その中の開き括弧と組になっている閉じ括弧が記述されている第5行目「}」が検索される。そして、その第5行目も、図16（B）に示すように選択状態とされる。

【0076】また、例えば、図17（A）に示すようなスクリプトにおいて、第3行目乃至第6行目が選択状態とされた場合には、各行について、その行と組になっている行が検索される。

【0077】即ち、第3行目「もしYESならば{」については、その中の開き括弧と組になっている閉じ括弧が記述されている第16行目「}」が検索される。

【0078】第4行目「もしYESならば{」については、その中の開き括弧と組になっている閉じ括弧が記述されている第10行目「}さもなくば{」が検索される。さらに、第4行目については、第10行目「}さもなくば{」が検索された結果、その中の開き括弧と組に

なっている閉じ括弧が記述されている第14行目「}」も検索される。

【0079】第5行目「式「\$n=0」を計算する」については、どの行とも組になっていないから、どの行も検索されない。

【0080】第6行目「もしYESならば{」については、その中の開き括弧と組になっている閉じ括弧が記述されている第8行目「}」が検索される。

【0081】そして、検索された行は、すべて選択状態とされ、その結果、図17（B）に示すように、第3乃至第6行目の他、第8行目、第10行目、第14行目、第16行目も選択状態とされる。

【0082】図17（B）に示した状態において、削除を指令すると、選択状態とされている行はすべて削除され、スクリプトは、図18に示すようになる。

【0083】以上のように、スマートスクリプトエディタでは、スクリプトのある行が選択されたときに、その行が選択状態とされる他、その選択された行と組になっている行も選択状態とされるので、その選択状態とされた部分を削除しても、その削除の結果得られるスクリプトにおいては、元のスクリプトの文法構造が正しければ、その正しい文法構造が維持される。その結果、ユーザによるスクリプトの編集ミスを減らし、作業効率を向上させることができる。

【0084】また、選択状態とされた部分も、その文法構造は正しいから、これを切り取って他のスクリプトに貼り付けたり、またはコピーしても、その他のスクリプトの文法構造が破壊される（誤った文法構造とされる）ことはない。

【0085】次に、図19のフローチャートを参照して、スマートスクリプトエディタにおいて、スクリプトの切り取り、コピー、削除を行うために、その切り取り、コピー、削除を行う行が、ユーザによって選択されたときの処理について説明する。

【0086】切り取り、コピー、削除を行う行が、ユーザによって選択されると、その選択された行が選択状態とされる。そして、ステップS11において、選択状態とされている行の、例えば最初の行（最も上部にある行）が、注目行として取得され、ステップS12に進み、注目行の中に、他の記述（例えば、閉じ括弧）と組になる記述（例えば、開き括弧）が含まれるかどうかが判定される。

【0087】ステップS12において、注目行の中に、他の記述と組になる記述が含まれないと判定された場合、ステップS13に進み、選択状態とされている行のうち、まだ、注目行とされていない行があるかどうかが判定される。

【0088】ステップS13において、選択状態とされている行のうち、まだ、注目行とされていない行があると判定された場合、ステップS14に進み、選択状態と

されている行のうち、まだ、注目行とされていない行が新たに注目行として取得され、ステップS12に戻り、以下、同様の処理が繰り返される。

【0089】また、ステップS12において、注目行の中に、他の記述と組になる記述が含まれていると判定された場合、ステップS15に進み、注目行が、関数定義の開き括弧「{」または閉じ括弧「}」のうちのいずれかを含むかどうかが判定される。

【0090】ここで、スマートスクリプトエディタにおいて扱われるスクリプトは、関数を用いて記述されるようになされており、関数の始まりには、開き括弧「{」が、関数の終わりには、閉じ括弧「}」が、それぞれ記述されるようになされている。

【0091】ステップS15において、注目行が、関数定義の開き括弧「{」または閉じ括弧「}」のうちのいずれかを含むと判定された場合、ステップS16に進み、その関数全体に亘る行すべてが、選択状態とされる。

【0092】また、ステップS15において、注目行が、関数定義の開き括弧「{」および閉じ括弧「}」のうちのいずれも含まない判定された場合、ステップS17に進み、注目行の記述と組になる記述がある行が選択状態にされる。

【0093】ステップS16およびS17の処理後は、いずれも、ステップS13に進み、以下、上述した処理が行われる。

【0094】一方、ステップS13において、選択状態とされている行のうち、まだ、注目行とされていない行がないと判定された場合、ステップS18に進み、選択状態とされている行を対象に、切り取り、コピー、または削除が行われ、処理を終了する。

【0095】なお、関数全体を選択状態にして、切り取りや削除を行う場合には、関数全体の切り取りまたは削除を行って良いのかどうかの確認をするためのダイヤログボックスを表示するようになることができる。

【0096】また、本実施の形態では、スクリプトの編集を対象としたが、他のプログラムの編集を対象に、上述したような処理を行うことも可能である。

【0097】さらに、本実施の形態では、組になる記述として、開き括弧と閉じ括弧とを用いたが、その他、あるブロックの開始と終わりを規定する記号が定義されている場合には、そのような記号を、組になる記述として扱うことが可能である。

【0098】次に、図20は、スマートスクリプトエディタにおいて、デスクトップ上にオープンされるウインドウを示している。

【0099】スマートスクリプトエディタにおいては、図20に示すように、図10に示したメインウインドウ131の他、コマンドツリーウインドウ132および変数ツリーウインドウ133がオープンされるようになされている。

【0100】メインウインドウ131は、メニューバー141、ツールバー142、編集ウインドウ143等から構成されている。メニューバー141においては、各項目において、各種の処理を行うためのメニューがプルダウン形式で表示されるようになされている。ツールバー142は、各種の操作を行うためのボタン151乃至168で構成されている。

【0101】即ち、ボタン151は、新たなスクリプトを編集するための編集ウインドウ143をオープンするときに操作され、ボタン152は、既にファイルとして記憶されているスクリプトを編集するための編集ウインドウ143をオープンするときに操作される。ボタン153は、スクリプトをファイルとして保存するときに操作され、ボタン154は、スクリプトの選択状態となっている部分を切り取るときに操作される。ボタン155は、スクリプトの選択状態となっている部分をコピーするときに操作され、ボタン156は、スクリプトの切り取った部分を貼り付けるときに操作される。ボタン157は、直前に行った操作を取り消すときに操作され、ボタン158は、その取り消した操作をやり直すときに操作される。ボタン159は、コマンドツリーウインドウ132の表示をオン／オフするときに操作され、ボタン160は、変数ツリーウインドウ133の表示をオン／オフするときに操作される。ボタン161は、編集ウインドウ143に表示されたスクリプトの日本語による表示をオン／オフするときに操作され、ボタン162は、アクティブになっている編集ウインドウ143に表示されたスクリプトを実行し、また、その実行を一時停止するときに操作される。ボタン163は、アクティブになっている編集ウインドウ143に表示されたスクリプトを、いわゆるステップ実行するときに操作され、ボタン164は、スクリプトの実行を停止するときに操作される。ボタン165は、ユーザによる操作手順を、スクリプトとして記録する場合において、その記録を開始するとき、および記録を終了するときに操作される。ボタン167は、スマートスクリプトエディタのバージョン情報を表示するときに操作され、ボタン168は、スマートスクリプトエディタのヘルプを表示するときに操作される。

【0102】編集ウインドウ143には、スクリプトが表示される。なお、編集ウインドウ143は、複数オープンすることができるようになされており、図20においては、2つの編集ウインドウ143がオープンされている。

【0103】コマンドツリーウインドウ132は、編集ウインドウ143にスクリプトを入力する場合に、その入力を支援するためのもので、そこには、スクリプトの記述に用いることのできるコマンドが、木構造で表示される。いま、このコマンドを木構造で表示したものコマンドツリーというものとすると、コマンドツリーにお

いっては、そのノードとなっている部分で、「+」の表示をクリックすると、図21に示すように、そのノードの概念に含まれるコマンドが表示される（ノードが開く）ようになっており、これにより、ユーザが、容易に、所望のコマンドを探すことができるようになっている。

【0104】さらに、コマンドツリーにおいては、例えば、ユーザが所望のコマンドをクリックすると、そのコマンドが、編集ウインドウ143に反映されるようになされており、これにより、ユーザが、容易に、スクリプトの記述を行うことができるようになされている。

【0105】また、開いたノードは、図21において「-」の表示をクリックすると、元の状態になる（閉じる）ようになっている。従って、必要なないコマンドのノードは閉じておくことで、コマンドツリーウインドウ132がデスクトップに占める面積を小さくすることができ、これにより、使用感を向上させることができようになっている。即ち、コマンドツリーウインドウ132を小さく表示することにより、コマンドツリーウインドウ132によって、他のウインドウの表示が遮られることを防止することができ、また、コマンドツリーウインドウ132を操作してから、メインウインドウ131を操作する場合に、カーソルの移動量が少なくて済むようになる。

【0106】変数ツリーウインドウ133は、スクリプトで使用される変数の取り扱いを支援するためのもので、そこには、スクリプトで使用される変数が、型ごとに分類され、木構造で表示される。いま、この変数を木構造で表示したものを変数ツリーというものとすると、変数ツリーは、図22に示すように、スクリプトで使用されている関数が最上位階層のノードとして構成される。なお、図22の実施の形態は、関数としては、main()だけが表示されているが、この関数main()は、スクリプトを記述するための編集ウインドウ143がオープンされると、自動的に設けられる。また、変数ツリーの最上位階層には、図22に示すように、関数のノードの他、便宜上、すべての関数から参照可能なグローバル変数のノードがあらかじめ設けられている。

【0107】変数ツリーにおいては、関数のノードを最上位階層として、各関数のノードの下位階層には、変数の各種の型のノードが設けられている。ここで、図22の実施の形態においては、スカラー変数型、ウインドウ変数型、ファイル変数型、スカラー配列変数型のノードが、各関数のノードの下位階層に設けられている。

【0108】さらに、変数の型のノードの下位階層には、その型の新規の変数を追加するための「新しい変数の追加」ノードが設けられている。この「新しい変数の追加」ノードがクリック（選択）されると、その「新しい変数の追加」ノードを下位階層とする型のノードに、その型の新規の変数が追加される。

【0109】即ち、図22における変数ツリーにおい

て、関数main()のノードの下位階層のスカラー変数型のノードの、さらに下位階層の「新しい変数の追加」ノードがクリックされると、例えば、図23に示すような、新規の変数（変数名）を入力するためのダイアログボックス171が表示される。このダイアログボックス171には入力欄172が設けられており、この入力欄172に、変数名を入力すると、変数ツリーにおいては、関数main()のノードの下位階層のスカラー変数型のノードの下位階層に、その変数名の変数が追加される。

【0110】例えば、いま、編集ウインドウ143において、図24に示すようなスクリプトが編集されている場合においては、変数ツリーウインドウ133には、図25に示すような変数ツリーが表示される。ここで、図25においては、上述したように、変数ツリーにあらかじめ設けられるグローバル変数および関数main()のノードの他に、関数func()が設けられているが、main()以外の関数は、図21に示したコマンドツリーにおいて、ユーザ定義関数のノードの下位階層にある所定のノード（図示せず）をクリックすることで、編集ウインドウ143に記述されるようになされている。

【0111】変数ツリーにおいても、コマンドツリーと同様に、ノードとなっている部分で、「+」の表示をクリックすると、図26に示すように、そのノードの下位階層のノードが開き、また、開いたノードは、「-」の表示をクリックすると閉じるようになっている。従って、変数ツリーによっても、ユーザが、容易に、所望の変数を探すことができ、さらに、ユーザの使用感を向上させることができようになっている。

【0112】ここで、図25の変数ツリーのスカラー変数のノードを開いた状態を示した図26の変数ツリーによれば、関数main()では、スカラ型の変数として、argc, x, \_resultが使用されることが分かる。なお、図26の変数ツリーは、図24に示した編集ウインドウ143に表示されたスクリプトについてのものであるが、このスクリプトの関数main()において、変数\_resultが表示されていないのは、この変数\_resultは、スマートスクリプトエディタが、いわば暗黙のうちに使用するものであるからである。また、スカラ型の変数argc, x, \_resultの先頭に付加されている\$は、それに続く変数（変数名）がスカラ型であることを表す。スマートスクリプトエディタにおいて編集されるスクリプトで使用される変数については、その型が、変数名の先頭に付加される記号で区別されるようになされており、具体的には、例えば、スカラ型の変数には\$が、ウインドウ型の変数には%が、スカラー配列型の変数には@が、それぞれ付加されるようになされている。

【0113】従って、上述したように、変数ツリーにおいて、「新しい変数の追加」ノードがクリックされると、図23に示したダイアログボックス171が表示され、その入力欄172に、変数名を入力すると、変数ツ

リーに、その変数が追加されるが、この追加される変数は、入力欄172に入力された変数の先頭に、その変数の型を表す記号（上述したような\$や%など）が付加されたものとなる。

【0114】また、上述したように、スマートスクリプトエディタでは、そのメインウインドウ131のボタン162（図20）をクリックすることで、編集ウインドウ143に記述されたスクリプトの実行が開始され、あるいはその実行が一時停止されるが、変数ツリーでは、スクリプトの実行が一時停止された場合に、その一時停止時における各変数に関する情報が、図27に示すように表示されるようになされている。

【0115】即ち、図27は、図24の編集ウインドウ143に表示されたスクリプトが実行され、その実行が一時停止されたときの変数ツリーを示している。

【0116】図27（A）は、関数func()のノードの下位階層のスカラー型のノードが開かれた状態を示しており、実行が一時停止されたときに、関数func()で使用されているスカラー型の変数nに、5がセットされていることが分かる。

【0117】図27（B）は、関数main()のノードの下位階層のスカラー配列型のノードと、ウインドウ変数型のノードが開かれた状態を示しており、実行が一時停止されたときの、関数main()で使用されているスカラー配列型の変数arrayにセットされている値と、ウインドウ型の変数nに関する情報が分かる。

【0118】ここで、上述したように、スカラー型の変数については、その変数にセットされている値が表示されるが、スカラー配列型の変数や、ウインドウ型の変数については、より詳細な情報が表示される。即ち、スカラー配列型の変数については、その変数の各配列要素（例えば、図27（B）におけるarray[0], array[1], array[2]）にセットされている値が表示され、ウインドウ型の変数については、Windows95によって設定されるウインドウハンドル（例えば、図27（B）における00030442）、ウインドウ名（ウインドウのタイトルバーの表示）、クラス名、コントロールID（ウインドウの親子関係を規定するため、子のウインドウに付されるもの）が表示される。

【0119】なお、例えば、前回行われた一時停止から、今回行われた一時停止の間に値の変更された変数については、変更のない変数と異なる表示、即ち、例えば、太字による表示が行われるようになされている（図27においては、変数n, array[0], array[1], array[2]が太字で表示されている）。これにより、ユーザーは、変更があった変数を、一目で認識することができるようになされている。

【0120】以上のように、変数ツリーにおいては、実行が一時停止されたときに、その一時停止時における変数に関する情報が表示されるので、この表示を参照する

ことで、例えば、スクリプトのデバック効率等を向上させることができる。

【0121】次に、スマートスクリプトエディタでは、変数ツリーに追加した変数は、例えば、その変数ツリー上の変数をクリック等することによって、編集ウインドウ143に入力することもできるが、その他、以下のように、プルダウンメニューから選択することによって入力することもできるようになされている。

【0122】即ち、上述したように、コマンドツリーによれば、そこに表示されているコマンドをクリックすると、そのコマンドを、編集ウインドウ143に入力することができるようになされているが、コマンドの中には、変数の指定を必要とするものがある。具体的には、例えば、図21のコマンドツリーにおいて、ウインドウのノードの下位階層の操作のノードの下位階層に属するコマンド「アクティブにする」は、所定のウインドウをアクティブにすることを指示するものであるが、このコマンド「アクティブにする」を用いる際には、アクティブにするウインドウを表す変数を指定する必要がある。

【0123】この場合、スマートスクリプトエディタでは、例えば、図28に示すような、アクティブにするウインドウを表す変数を入力するためのダイアログボックス181が表示される。このダイアログボックス181においては、入力欄182が設けられており、この入力欄182に、アクティブにするウインドウを表す変数を、直接入力することも可能であるが、ボタン182Aをクリックすることで、変数ツリーに追加されたウインドウ型の変数が、プルダウンメニュー形式で表示されるようになされている。

【0124】即ち、スマートスクリプトエディタでは、コマンド「アクティブにする」が記述される関数のノードのウインドウ型のノードの下位階層に属するウインドウ型の変数が、変数ツリーから選択され、プルダウンメニュー形式で表示される。

【0125】この場合、ユーザーは、プルダウンメニューに表示された変数をクリックするだけで、アクティブにするウインドウを表す変数を、入力欄182に入力することができ、その結果、編集ウインドウ143に対する変数の入力を、容易に行うことができる。

【0126】以上のように、変数が、型ごとに分類され、木構造で表示されるので、スクリプトで使用される変数の取り扱いを容易にすことができ、その結果、スクリプトの記述を、容易に行うことが可能となる。

【0127】次に、図29のフローチャートを参照して、スマートスクリプトエディタによる変数ツリーに関する処理について説明する。

【0128】カーソル等の操作により、変数ツリーに関するイベントが生じると、スマートスクリプトエディタでは、ステップS21において、そのイベントが解析され、ステップS22に進む。ステップS22では、その

イベントの解析結果にしたがって、所定の処理が行われ、処理を終了する。

【0129】即ち、例えば、ボタン160(図20)がクリックされた場合には、変数ツリーウィンドウ133がオープンされ、変数ツリーが表示され、あるいは、変数ツリーウィンドウ133がクローズされる。また、例えば、変数ツリーにおける「新しい変数の追加」ノードがクリックされた場合には、変数ツリーに、新規の変数が追加される。さらに、例えば、ボタン182A(図28)がクリックされた場合には、変数ツリーから所定の変数が選択され、フルダウンメニュー形式で表示される。また、例えば、スクリプトが実行されている場合に、ボタン162(図20)が操作されることにより、その実行の一時停止が指令されたときには、その一時停止が指令されたときの変数に関する情報が、変数ツリーに表示される。

【0130】なお、上述の場合においては、スクリプトに記述される変数について、変数ツリーを構成するようにしたが、変数ツリーは、その他のプログラムに記述される変数について構成することも可能である。

【0131】さらに、変数ツリーは、ユーザが、例えばテキスト等で記述したスクリプト等について構成することも可能である。

【0132】また、本明細書において、上述した実行エンジンやスマートスクリプトエディタとして機能するコンピュータプログラムをユーザに提供する提供媒体には、磁気ディスク、CD-ROMなどの情報記録媒体の他、インターネット、デジタル衛星などのネットワークによる伝送媒体も含まれる。

【0133】

【発明の効果】以上の如く、本発明の編集装置および編集方法、並びに提供媒体によれば、編集対象の記述の少なくとも一部が選択されたときに、その一部分が選択状態とされ、その選択された編集対象の記述の一部と組になっている部分が、編集対象の記述から検索される。そして、編集対象の記述から検索された部分も選択状態とされる。従って、所定の文法構造を有する記述を対象とした編集を行う場合において、正しい文法構造を維持することが可能となる。

【図面の簡単な説明】

【図1】従来の、スクリプトを編集するエディタの編集画面を示すディスプレイ上に表示された中間階調の写真である。

【図2】図1の編集画面の中の、スクリプトが記述された状態のスクリプト編集ウィンドウを示すディスプレイ上に表示された中間階調の写真である。

【図3】本発明を適用したパソコン1の構成例を示す外観斜視図である。

【図4】図3の平面図である。

【図5】図3の表示部を本体に対して閉塞した状態を示

す左側側面図である。

【図6】図3の表示部を本体に対して180度開いた状態を示す右側側面図である。

【図7】図5の正面図である。

【図8】図6の底面図である。

【図9】図3のパソコン1の電気的構成例を示すブロック図である。

【図10】スマートスクリプトエディタのメインウィンドウ131を示すディスプレイ上に表示された中間階調の写真である。

【図11】Windows95のデスクトップを示すディスプレイ上に表示された中間階調の写真である。

【図12】タスクバー103を示すディスプレイ上に表示された中間階調の写真である。

【図13】アイコン表示処理を説明するためのフローチャートである。

【図14】タスクトレイ114を示すディスプレイ上に表示された中間階調の写真である。

【図15】インタプリタが起動されたときに表示されるダイアログボックス121を示すディスプレイ上に表示された中間階調の写真である。

【図16】スクリプトの例を示すディスプレイ上に表示された中間階調の写真である。

【図17】スクリプトの例を示すディスプレイ上に表示された中間階調の写真である。

【図18】スクリプトの例を示すディスプレイ上に表示された中間階調の写真である。

【図19】削除等を行う行が、ユーザによって選択されたときに行われる処理を説明するためのフローチャートである。

【図20】スマートスクリプトエディタがにおいて表示されるウインドウを示すディスプレイ上に表示された中間階調の写真である。

【図21】コマンドツリーウィンドウ132を示すディスプレイ上に表示された中間階調の写真である。

【図22】変数ツリーウィンドウ133を示すディスプレイ上に表示された中間階調の写真である。

【図23】新規の変数を追加するときに表示されるダイアログボックス171を示すディスプレイ上に表示された中間階調の写真である。

【図24】スクリプトの例を示すディスプレイ上に表示された中間階調の写真である。

【図25】変数ツリーを示すディスプレイ上に表示された中間階調の写真である。

【図26】変数ツリーを示すディスプレイ上に表示された中間階調の写真である。

【図27】変数ツリーを示すディスプレイ上に表示された中間階調の写真である。

【図28】変数を入力するときに表示されるダイアログボックス181を示すディスプレイ上に表示された中間

階調の写真である。

【図29】変数ツリーに関して行われる処理を説明するためのフローチャートである。

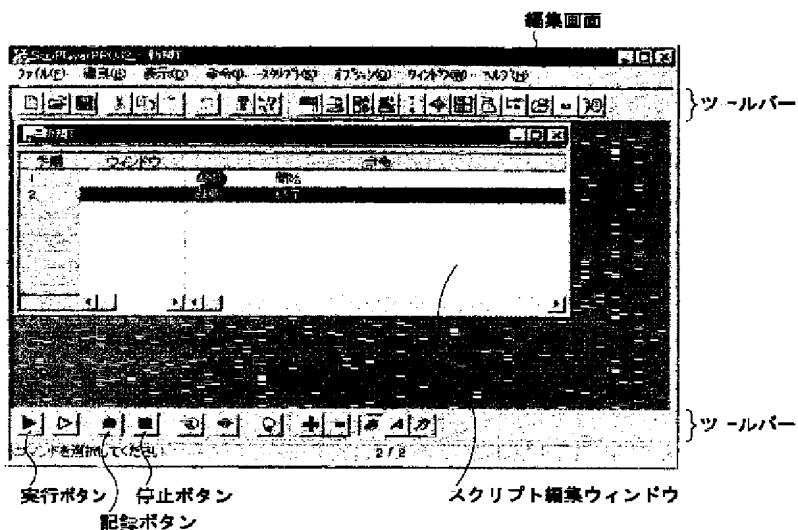
【符号の説明】

1 パーソナルコンピュータ, 2 本体, 3 表示部, 52 CPU, 54 RAM, 56 HDD, 101 ウィンドウ, 102 開くボタン, 103 タスクバー, 111 スタートボタン, 112 アイコン, 113 タスクボタン, 114 タスクトレイ, 114A アイコン, 121 ダイアログボ

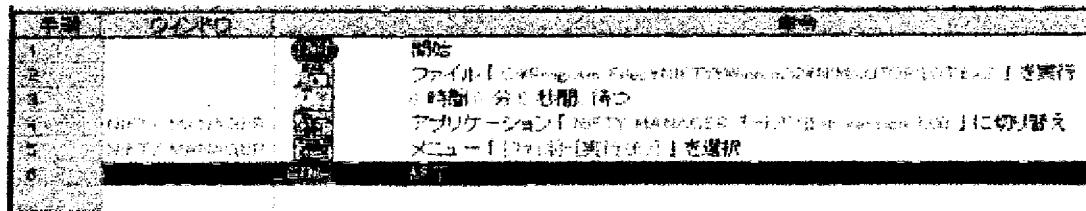
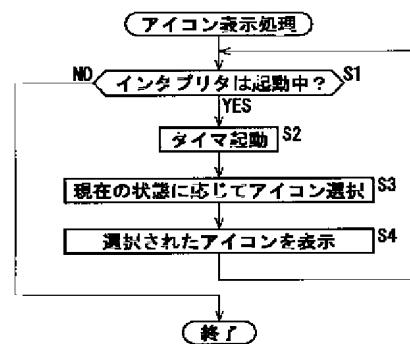
ックス, 121A アイコン, 122 最小化ボタン, 123 クローズボタン, 124 キャンセルボタン, 131 メインウインドウ, 132 コマンドツリー ウィンドウ, 133 変数ツリー ウィンドウ, 141 メニューバー, 142 ツールバー, 143 編集ウインドウ, 151 乃至 168 ボタン, 171 ダイアログボックス, 172 入力欄, 181 ダイアログボックス, 182 入力欄, 182A ボタン

【図1】

【図13】



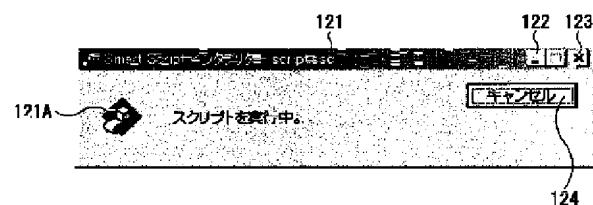
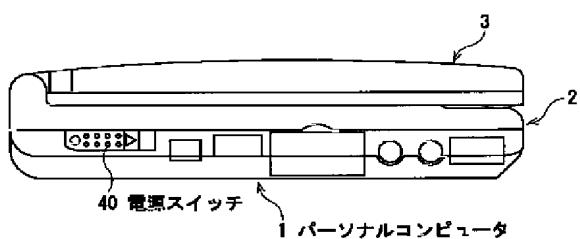
【図2】



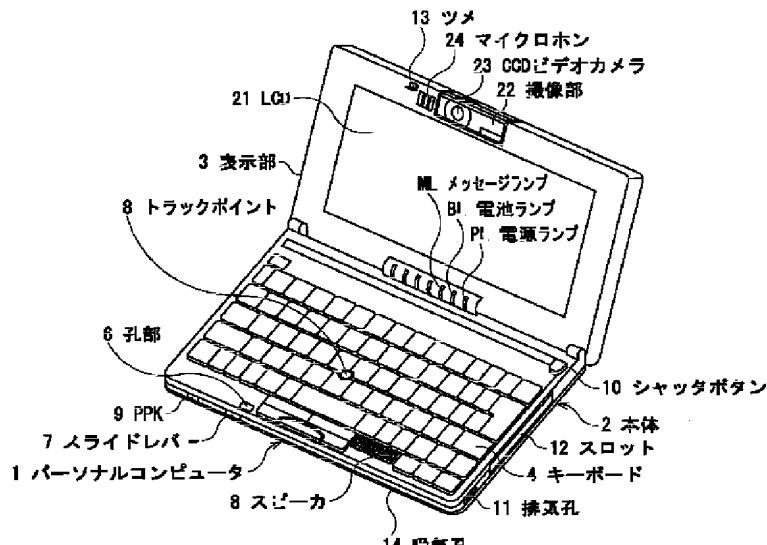
スクリプト編集ウィンドウ

【図5】

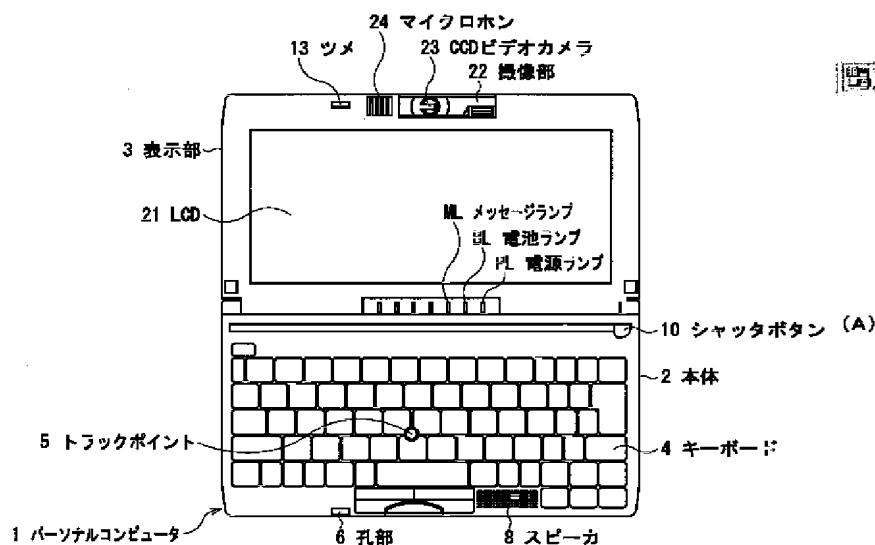
【図15】



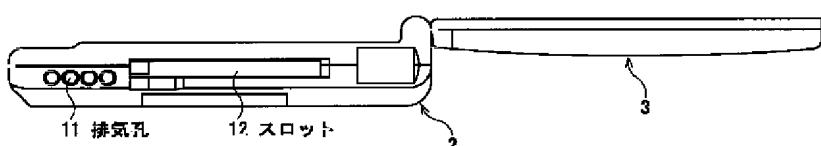
【図3】



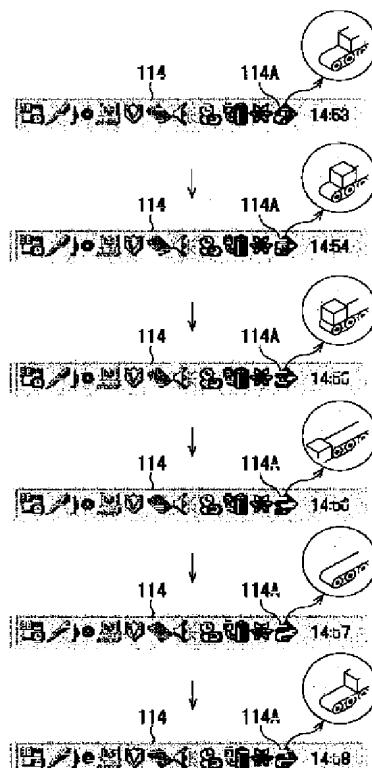
【図4】



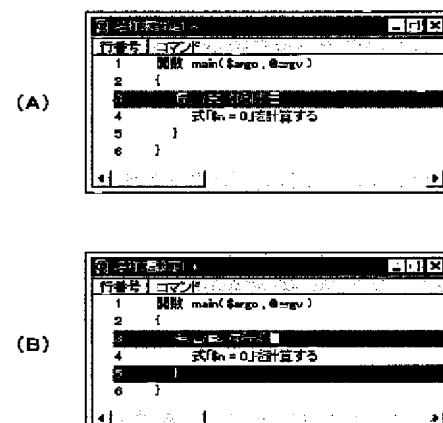
【図6】



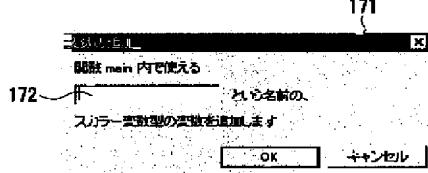
【図14】



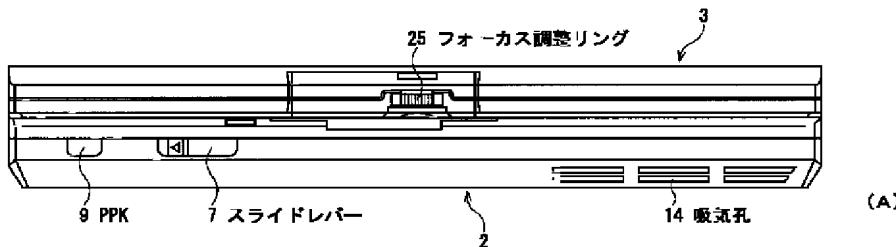
【図16】



【図23】

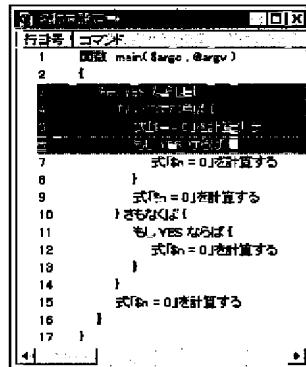


【図7】

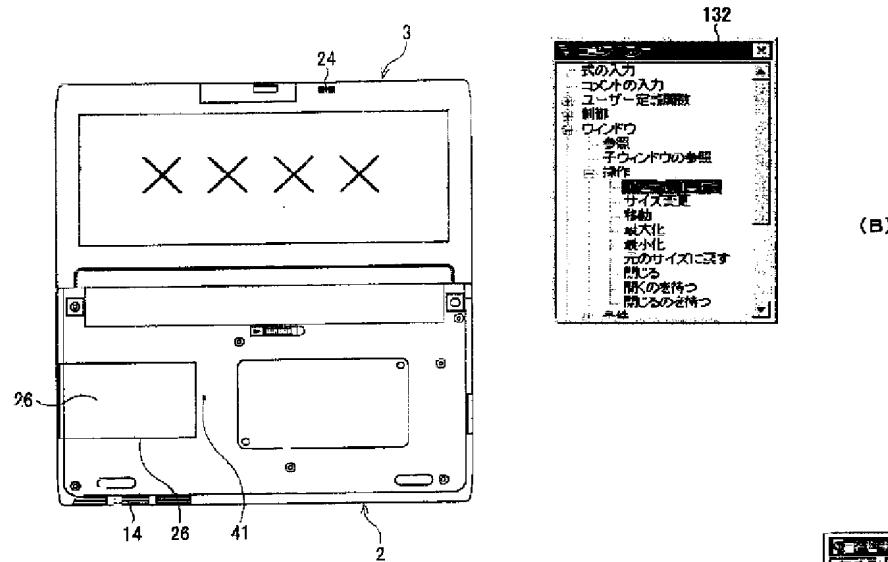


[図8]

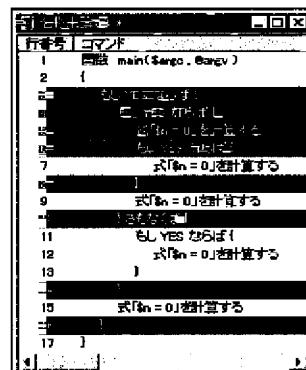
〔図21〕



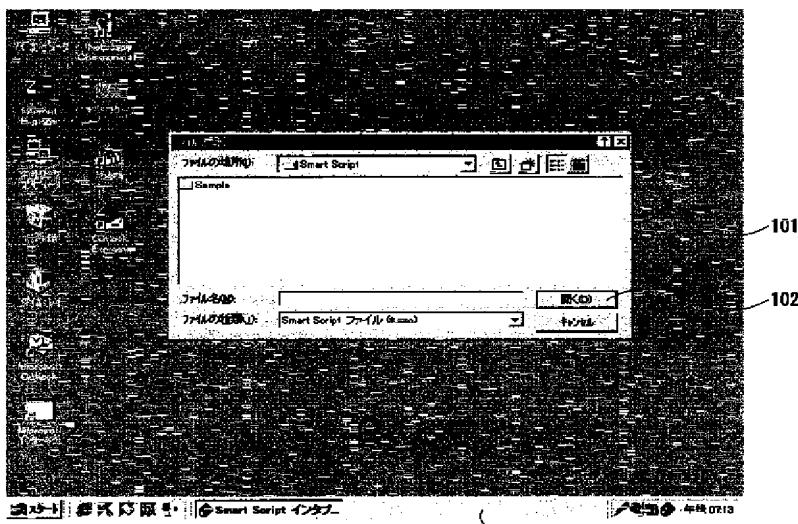
【図17】



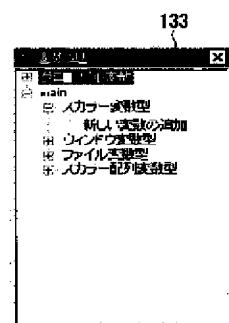
### 【圖 1-1】



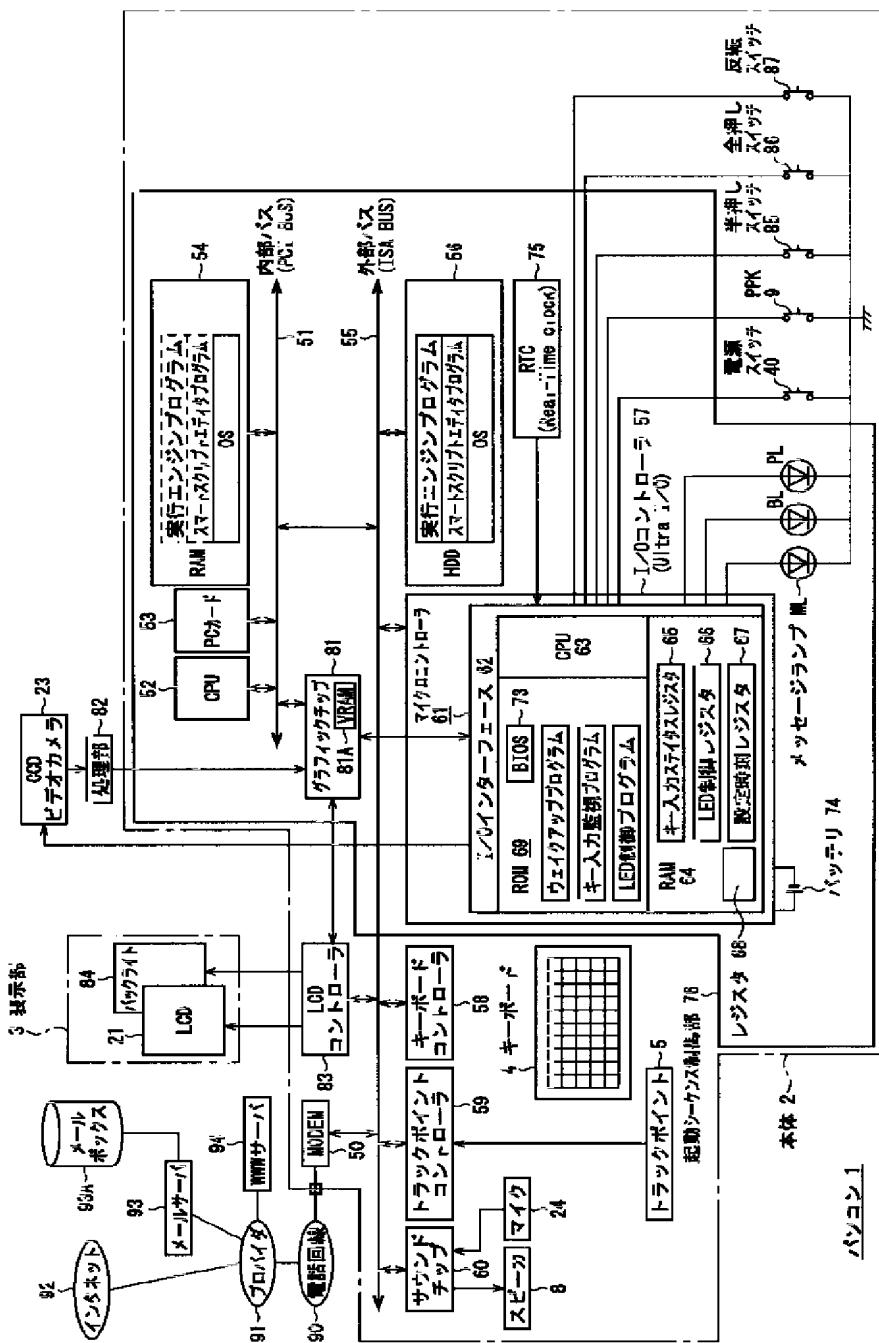
【図18】



【図22】



【図9】



【図10】

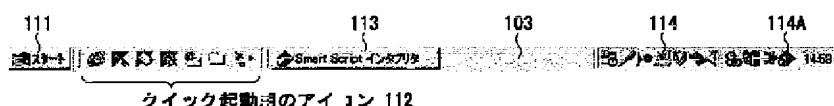
131

```

6 もし NO ならば {
7 1'を終了コードとしてスクリプト終了
8 }
9 メッセージ「メモ帳を使用して説明します。注意:現在開集中のメモ帳が複数個あります。」を表示
10 ウィンドウ(「* - メモ帳」, 「Notepad」)は存在する?
11 もし YES ならば {
12 メッセージ「現在実行中のメモ帳が存在します。このスクリプトを実行すると、1'を終了コードとしてスクリプト終了」を表示
13 }
14 メッセージ「メモ帳を起動し、開くのを待ち、ウィンドウハンドルを%notepad%ファイル(「notepad」)を実行する」を表示
15 //ウィンドウが開くのを待つ
16 ウィンドウ(「無題 - メモ帳」, 「Notepad」)が開くのを最大15秒待ち、ウィンドウ変数[%notepad]は有効?
17 もし YES ならば {
18 メッセージ「開くことができました。開くのを待つ」命令で成功した場合、1'をもくは【
19 }
20 ヘルプ表示するには [F1] を押してください。

```

【図12】



【図24】

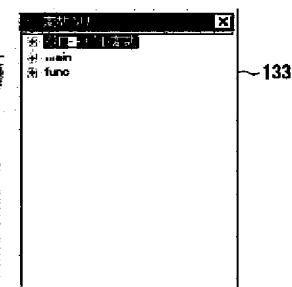
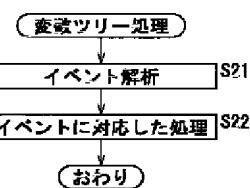
143

```

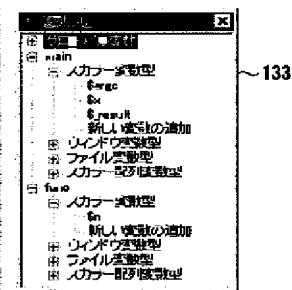
1 関数 main($args, $args)
2 1
3 ユーザー選択肢を出し $x = func(5)
4 メッセージ「$x」ボタン(「OKボタン」)のメッセージボックスを表示する
5 //
6 関数 $ func($n)
7 {
8 もし式「$n > 1」が成立立つならば {
9 $t($n = $n - 1)を計算する
10 1
11 $tを出力値として戻値から取得
12 }
13 }

```

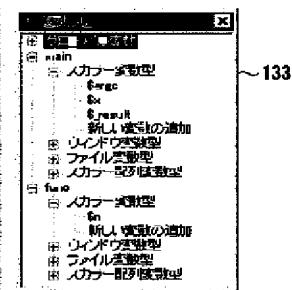
【図29】



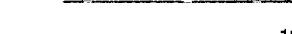
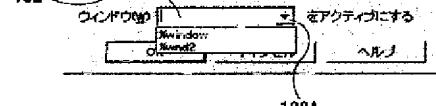
【図25】



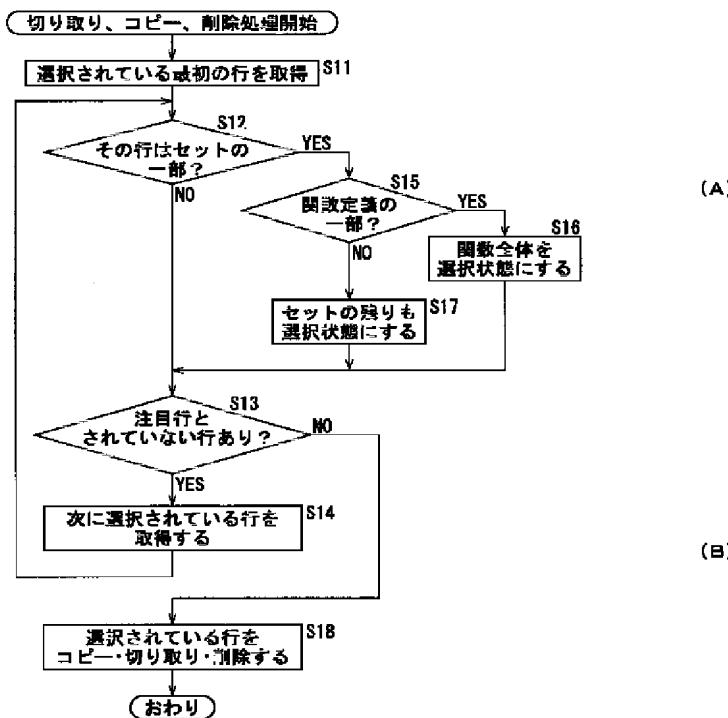
【図26】



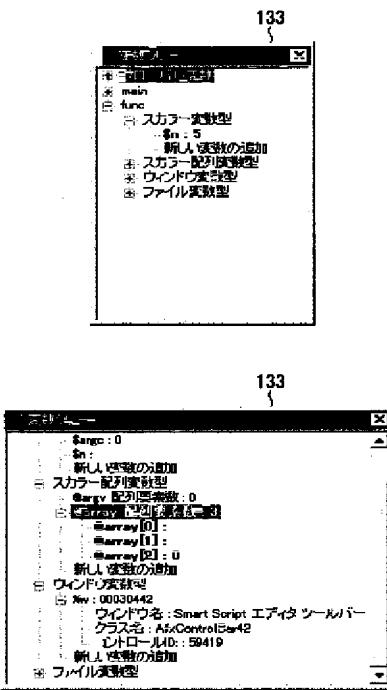
【図28】



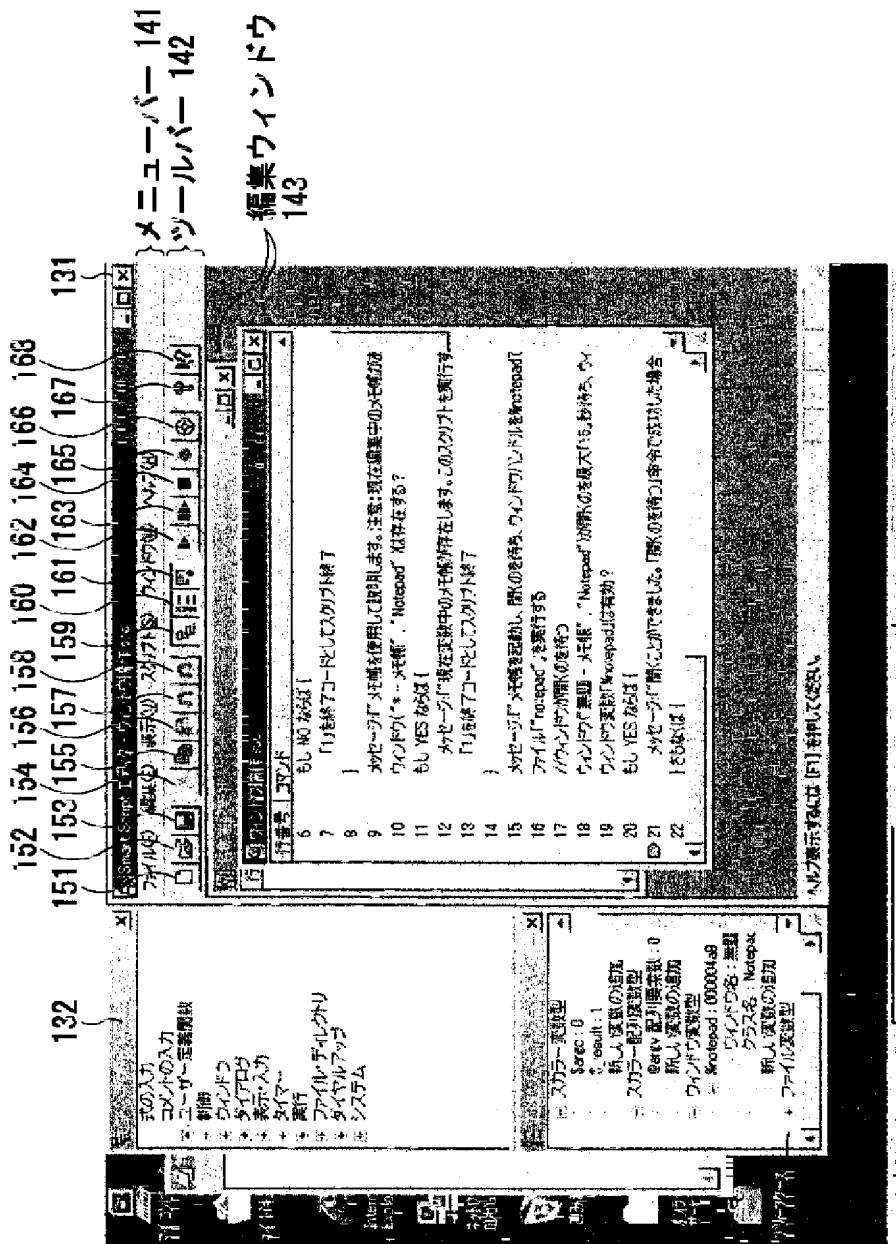
【図19】



【図27】



【図20】



## フロントページの続き

(72)発明者 桑折 隆之  
東京都品川区北品川6丁目7番35号 ソニ  
一株式会社内

(20)00-222190 (P2000-222190A)

Fターム(参考) 5B076 DA04 DC02 DC04  
5E501 AA03 AC33 AC34 BA05 CA04  
CB08 EA05 EA10 EA11 EA12  
EA13 EB05 EB12 EB13 EB14  
EB17 FA04 FA05 FA06 FA15  
FB28